Available online at www.sciencedirect.com
ScienceDirect
Computers and Mathematics with Applications 55 (2008) 2746–2752
An International Journal
computers & mathematics
with applications
www.elsevier.com/locate/camwa

A vector partitioning approach to detecting community structure in complex networks☆

Gaoxia Wang, Yi Shen*, Ming Ouyang

Department of Control Science and Engineering, Huazhong University of Science and Technology, Wuhan, 430074, PR China

Received 26 March 2007; received in revised form 24 September 2007; accepted 10 October 2007

Abstract

In recent years, the problem of community structure detection has attracted more and more attention and many approaches have been proposed. Recently, Newman pointed out that this issue can be transformed into the problem of constrained maximization of the assignment matrix over possible divisions of a network. He presents further that this maximization process can be written in terms of the eigenspectrum of the "modularity matrix". On the basis of this work and the vector partition approach in computer science, we propose a kind of multiway division approach for detecting community structure of complex networks. Experimental results indicate that the algorithm works well and is effective at finding both good communities and the appropriate number of communities.

Keywords: Complex networks; Community structure; Eigenspectrum; Modularity matrix; Vector partition approach

1. Introduction

Many systems can usefully be represented as networks or graphs—collections of vertices joined in pairs by edges. Examples include the Internet and the World Wide Web, citation networks, social networks, and biological and biochemical networks of various kinds [1–3].

The problem of detecting and characterizing the community structure is one of the outstanding issues [4–7]. There are different formal definitions of community structure; the most popular one is based on the relative frequency of links. In this case communities are seen as groups of nodes within which connections are dense and between which connections are sparser [8,9]. The community structure has been empirically found in many real technological, biological and social networks. They are believed to play an important role in the functional properties of complex network structure, and so detecting such structure could be of significant practical importance. For instance, in information networks, detecting the community structure allows one to mine information in a more efficient way, narrowing the exploring of a network as large as the World Wide Web to a limited portion of it [10]. When used

☆ This work was funded by the National Natural Science Foundation of China under Grant No. 60574025.
* Corresponding author.
E-mail addresses: gaoxiawang@163.com (G. Wang), yishen64@163.com (Y. Shen), pandasjtu@126.com (M. Ouyang).

0898-1221/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2007.10.028

in the analysis of large collaboration networks, such as companies or universities, communities reveal the informal organization and the nature of information flows through the whole system [11,12].

The problem of finding communities is closely related to the problem of graph partitioning in computer science and hierarchical clustering in sociology [13]. The traditional algorithms that come from the two subjects provided the inspiration for detecting the community structure. We can find impressions of these classical algorithms more or less from the approaches adopted in recent years for detecting community structure. In [13,14], Newman and Boccaletti et al. look back on the above traditional approaches, review some methods newly proposed in the physics community, and compare the performances of different algorithms. Here, we will give more consideration of the algorithms based on spectral analysis.

Previous approaches to graph partitioning from spectral analysis have been mostly developed in the computer science community. Spectral analysis provides a tool for bi-partitioning. Many networks contain more than two communities, so we would like to extend the method to find good divisions of networks into large numbers of parts. In practice, most approaches to graph partitioning have been based on iterative bisection. When applied to finding community structures these methods have the disadvantage that repeated bisection is not guaranteed to reach the best or most natural partition in general cases. Moreover, they give no indication of when the bisection should terminate, and thus need extra information on the expected number of communities [15].

This raises a new problem: How do we know when the communities found by the algorithm are good ones? To get round this problem, Newman introduces a benefit function "modularity", often using the symbol Q to represent it. The modularity is the number of edges falling within groups minus the expected number in an equivalent network with edges placed at random. This quantity, it is claimed, is high for good community divisions and low for poor ones. Thus, one can search for community structure precisely by looking for the divisions of a network that have positive, and preferably high, values of the modularity. It can also be used to automatically select the optimal number of communities p, by finding the value of p for which Q is maximized. In his recent paper [15,16], Newman uses the modularity matrix and the assignment matrix to represent the modularity. He shows that the problem of multiway division of community structure can be transformed into the problem of constrained maximization of the assignment matrix, and can be expressed in the form of Eq. (1) (see details in Section 2).

On the basis of the concept of modularity, some authors propose the spectral method for dividing networks into more than two communities.

White and Smyth apply the standard technique of clustering based on group centroids to this optimization problem and find good results. They project the network nodes into an eigenvector space of tunable dimensionality. Then they apply clustering techniques such as the k-means clustering method to split the network into various groups, calculate the modularity of possible groupings for every dimension considered for the eigenvector space and, finally, find the global maximum over all possible numbers of eigenvectors [17]. Doneti et al. adopt a method which combines spectral methods with hierarchical clustering techniques, and also use the modularity to develop building up the dendrogram and judge the results of division [18,19]. By introducing modularity, these new methods are flexible enough to apply to quite general network structure.

Our aim in this paper is to develop the spectral based algorithm to reveal the structure of a complex network. By analogy with Newman's method for reformulating the expression of the modularity using the eigenvectors and eigenvalues of the modularity matrix, we transform the problem into the vector partition question, and propose a new vector partition algorithm.

The method described in this paper all assume that we are given a network structure that we wish to divide into communities in such a way that every vertex belongs to one of the communities. We assume that the network is of the simplest kind possible, with a single type of undirected, unweighted edge connecting unweighted vertices of a single type.

2. Eigenvector space of the modularity matrix and community structure

Let $G(V, E, A)$ be an undirected network consisting of the set of vertices V, the set of edges E, and a symmetric adjacency matrix $A \in R^{n\times n}$, whose elements A_{ij} are equal to 1 if i points to j and 0 otherwise. We denote as d_i the degree of vertex i. Consider the partition of a network G into p non-overlapping communities. The corresponding assignment matrix $X = (x_{ih})$ has $x_{ih} = 1$ if vertex i belongs to community h and $x_{ih} = 0$ otherwise. In fact X represents the solution of our problem. We can generalize its properties as follows:

1. X is a 0–1 matrix; each row is non-zero and sums to unity. Column h has sum $|C_h|$, where $|C_h|$ denotes the number of vertices in the community h.
2. $XI_p = I_n$, where I_n is n ranks unit vector.
3. The columns of X are mutually orthogonal. The matrix satisfies the normalization condition $\text{Trace}(X^{\mathrm{T}}X) = n$. Here the trace of a matrix is the sum of its diagonal entries.

According to [15], modularity function Q can be expressed as $Q = \frac{1}{2m}\,\text{Trace}(X^{\mathrm{T}}BX)$, and the problem of the multiway community detection can be expressed as

$$\max_x\{\text{Trace}(X^{\mathrm{T}}BX)\} \text{ s.t. Trace } (X^{\mathrm{T}}X) = n \tag{1}$$

where B is the real symmetric matrix having elements $B_{ij} = A_{ij} - P_{ij}$, $P_{ij} = \frac{d_i d_j}{2m}$. This matrix is called the modularity matrix and it plays a role in the maximization of the modularity equivalent to that of the Laplacian in standard spectral partitioning. Note that the modularity matrix can take other forms [15]; we adopt here the simplest one.

The standard spectral partitioning methods are often based on analysis of the adjacency matrix A. In particular, such methods analyze simple functions of A, such as the Laplacian matrix $L = D - A$ and the normal matrix $N = D^{-1}A$ or $D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, where D is the diagonal matrix with elements $d_{ii} = \sum_{j=1}^{n} A_{ij}$ and n is the number of vertices in the network [20].

The Laplacian matrix has many attractive properties that make it very useful for partitioning large sparse arrays. L has the lowest eigenvalue 0 with eigenvector $(1, 1, \ldots, 1)$. When the graph consists of p components, i.e. it can be perfectly separated into p non-overlapping communities, the multiplicity of the null eigenvalue is equal to the number of components p. On the other hand, when the graph has an apparent, though not perfect, community structure with p communities, there will be only one eigenvalue equal to 0, $p - 1$ eigenvalues slightly larger than 0, and the remaining eigenvalues lying a gap away from 0 [10].

It is clear that there is useful information about the structure of network stored in the eigenvectors corresponding to the eigenvalues slightly larger than 0 of the Laplacian matrix [16].

Similar information is encoded in the modularity matrix. Analogously to the Laplacian matrix, the modularity matrix B always has a trivial eigenvalue equal to 0 and a corresponding trivial constant eigenvector, due to each row summing to zero. Unlike the case for the Laplacian, however, the eigenvalues of the modularity matrix are not necessarily all of one sign and in practice the matrix usually has both positive and negative eigenvalues [15]. In a network with an apparent community structure with p communities, B has also a certain number $p - 1$ of leading eigenvalues apparently larger than the other eigenvalues. So just as is the case with the Laplacian, the eigenspectrum of the modularity matrix is in general closely tied to the community structure.

In [18], Donetti et al. illustrate the relationship between the community structure and the eigenspectrum of the Laplacian matrix. This suggested to us applying this idea to the modularity matrix. We can choose the first few leading eigenvectors of the modularity matrix (note that they are not the first few non-trivial eigenvectors now); then each vertex in the graph can be represented by a point in a corresponding dimensional space in which the coordinates are given by projection on these eigenvectors. Fig. 1(a) shows the components of the first leading eigenvector of the modularity matrix of a computer-generated graph including 4 communities, each composed of 32 vertices (see Section 5 for details). We can identity the two communities clearly. Now taking into account some more eigenvectors, we can extract more precise information. This is illustrated in Fig. 1(b) and Fig. 1(c), where the nodes of the same graph are plotted using the components on the first two and three leading eigenvectors as coordinates, respectively. Simple inspection by eye shows that in Fig. 1(b), we can identify three communities clearly and in Fig. 1(c), all communities are distinctly separated now.

From above observation, we can see that if we want to detect the community structure clearly and directly from the eigenvectors of the modularity matrix, we should provide enough eigenvectors. If we use only a single eigenvector, we will find at most two groups. If we use two, we will find at most three groups, and so on. So the choice of how many eigenvectors p to work with is determined to some extent by the network: if the overall optimum modularity is for a division into p groups, we will certainly fail to find that optimum if we use less than $p - 1$ eigenvectors [15].

From another point of view, we now consider the assignment matrix X. Apart from the basic constraint conditions on the rows and columns of matrix X, the choice of p communities would be equivalent to choosing $p-1$ independent, mutually orthogonal columns $x_1, \ldots, x_{p-1}$. So the optimal modularity would be achieved by choosing exactly as

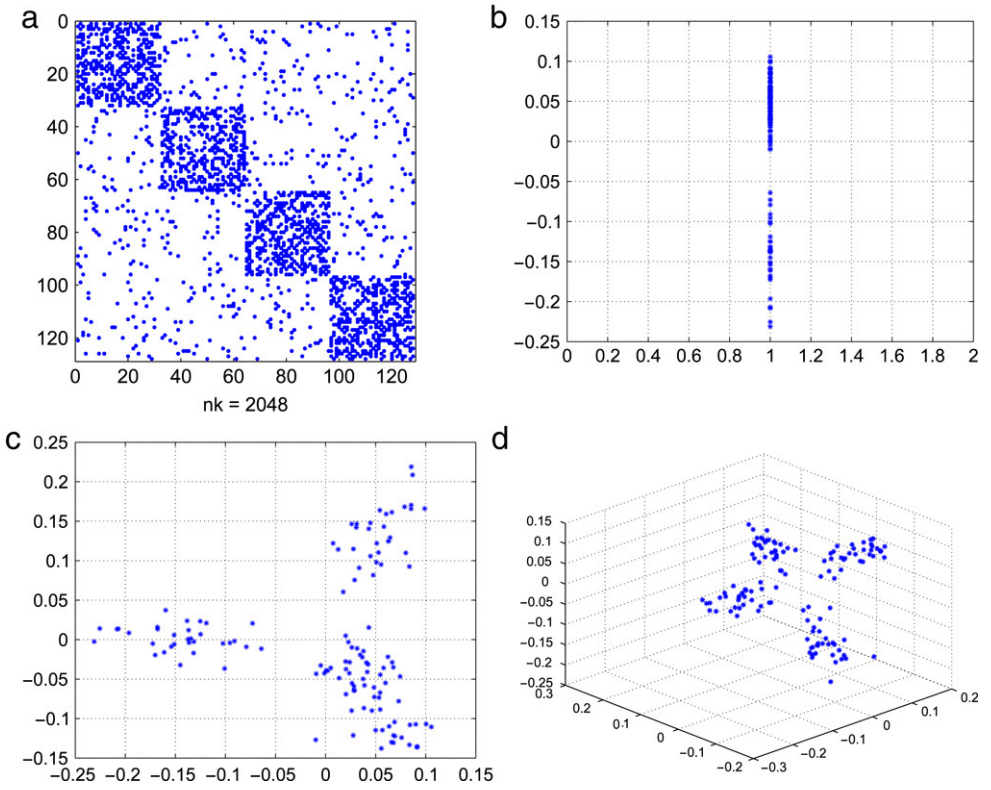

Fig. 1. (a) A computer-generated network with 128 nodes grouped into 4 communities. $k_{\text{in}} = 12$, $k_{\text{out}} = 4$. (b) Components of the leading eigenvector of the modularity matrix in (a). Two communities are clearly identified while the other two overlap. (c) Components of the first two positive eigenvectors. Three communities can be clearly identified. (d) All communities can be clearly identified when the components of the three most positive eigenvectors are plotted.

many independent columns of X as there are positive eigenvalues or, equivalently, by choosing the number of groups p to be 1 greater than the number of positive eigenvalues. Thus again we see that there is an intimate connection between the properties of the modularity matrix and the community structure of the network that it describes [15].

3. The description of the algorithm

We denote the set of normalized eigenvectors of B by $u_1, u_2, \ldots, u_n$ with corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. Without loss of generality, assume that the eigenvalues are labeled in decreasing order $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$, i.e. for $i \leqslant 1 \leqslant n$, $u_i^{\mathrm{T}} u_i = \|u_i\|^2 = 1$, $Bu_i = \lambda u_i$. Assume that matrix B has k positive eigenvalues and form the $k \times k$ eigenvalue matrix $\Lambda = (\lambda_{ii})$ having diagonal entries $\lambda_{ii} = \lambda_i$ and 0 entries elsewhere. There is a corresponding $n \times k$ eigenvector matrix $U_k = (u_1 \mid u_2 \mid \cdots \mid u_k)$.

As B is a real symmetric matrix, there exists a normal matrix U s.t. $B = U\Lambda U^{\mathrm{T}}$. So

$$Q = \mathrm{Trace}(X_{np}^T B_{nn} X_{np}) \tag{2}$$

$$= \mathrm{Trace}(X_{np}^T U_{nn} \Lambda_{nn} U_{nn}^T X_{np}). \tag{3}$$

Now, we only choose the p most positive eigenvalues in Eq. (2) and also drop the last $n - p$ eigenvectors of U; we have

$$\begin{aligned} Q &\simeq \mathrm{Trace}(X_{np}^T U_{np} \Lambda_{pp} U_{np}^T X_{np}) \\ &= \mathrm{Trace}(X_{np}^T U_{np} \Lambda_{pp}^{\frac{1}{2}} \Lambda_{pp}^{\frac{1}{2}} U_{np}^T X_{np}) \\ &= \mathrm{Trace}((\Lambda_{pp}^{\frac{1}{2}} U_{nP}^T X_{np})^T (\Lambda_{pp}^{\frac{1}{2}} U_{nP}^T X_{np})). \end{aligned}$$

We denote as y_i row i of $U\Lambda^{1/2}$, i.e.

$$y_i = (\sqrt{\mu_1}U_{i1}, \sqrt{\mu_2}U_{i2}, \ldots, \sqrt{\mu_k}U_{ik})$$

and now,

$$Q = \sum_{h=1}^{p} \|Y_h\|^2, \quad \text{here } Y_h = \sum_{i \in C_h} y_i \tag{4}$$

where C_h is the set of vertices comprising group k and the community vector Y_h, $h = 1, \ldots, p$.

The community structure detecting problem is now equivalent to choosing a division of the vertices into groups so as to maximize the magnitudes of the vectors Y_h. This means that we need to arrange that the individual vertex vectors y_i going into each group point in approximately the same direction. Problems of this type are called vector partitioning problems [21].

Eq. (4) shows that if y_i and y_j sum to a vector of large magnitude, then these vectors likely belong to the same subset. On the other hand it is intuitive that the vectors in a given group will point in the same general direction. This reminds us of a simple method for finding the maximum of modularity Q.

Now we take into account the simplest situation, i.e. we only choose the most positive eigenvector of the modularity matrix, and intend to find two possible communities. In this case, y is a one-dimensional vector, and each y_i is a point on a line. So just as in the standard spectral bisection method, we can separate the vertex by seeing whether the corresponding element in y is greater than or less than 0. In other words, we can get the two groups according to the sign of y_i.

Generally speaking, when we drop the first p positive eigenvectors, we can treat each y_i as a point in R^p, then cluster them into 2^p clusters according to the sign of the coordinate of y_i. Through this step, we get a coarse assignment matrix X^*. Then we choose the p groups which contain more vectors. Once we have p groups which contain some of the vectors, we should assign the rest of the vectors to these selected groups. Observe that adding a vector y_i to a community h produces a change in the corresponding term $\|Y_h - y_i\|^2 - \|Y_h\|^2 = \|y_i\|^2 + 2Y_h \cdot y_i$. Hence, we can increase the modularity by adding vertices for which $Y_h \cdot y_i > 0$.

4. Algorithm

Assume that we are seeking up to a maximum of k clusters ($k \leqslant p + 1$; p is the number of positive eigenvectors of modularity matrix B) and that we have an adjacency matrix A:

1. Compute the modularity matrix B.
2. Compute the first leading $k - 1$ eigenvalues and corresponding eigenvectors using a sparse eigenvector decomposition method. These eigenvectors form the matrix $U_k = (u_1, u_2, \ldots, u_{k-1})$.
3. For each value of h, $2 \leqslant h \leqslant k$:
   (a) Form the matrix U_h from the first $h - 1$ columns of U_k.
   (b) Compute $V = U_h\Lambda^{1/2}$. Set $V = \{y_1^h, y_2^h, \ldots, y_n^h\}$, where y_i^h is row i of V.
   (c) For each y_i^h, according to the signs of the elements of the vector, we can get an h-dimensional vector s_i. Its elements are 1 or -1, which is achieved by setting
   $$S_{ij}^h = \begin{cases} 1, & \text{if } y_{ij}^h \geqslant 0; \\ -1, & \text{if } y_{ij}^h < 0. \end{cases} \tag{5}$$
   (d) Construct a coarse assignment matrix X^*; put the y_i into 2^{h-1} groups.
   (e) Choose the first h groups which contain more vectors. Assume that the number of the remaining vectors is m; for i from 1 to m and j from 1 to h, calculate $Y_j \cdot y_i$, then find the maximum of these m values; finally add the y_i to the corresponding groups.
   (f) If $y_i \in C_h$ label node i as community h. So we get the assignment matrix X; x_{ij} equals 1 means node i is attributed to community j.
4. Calculate $Q(P^h) = \frac{1}{2m}\text{trace}(X^{\mathrm{T}}BX)$. Pick the h and the corresponding partition that maximizes $Q(P^h)$.

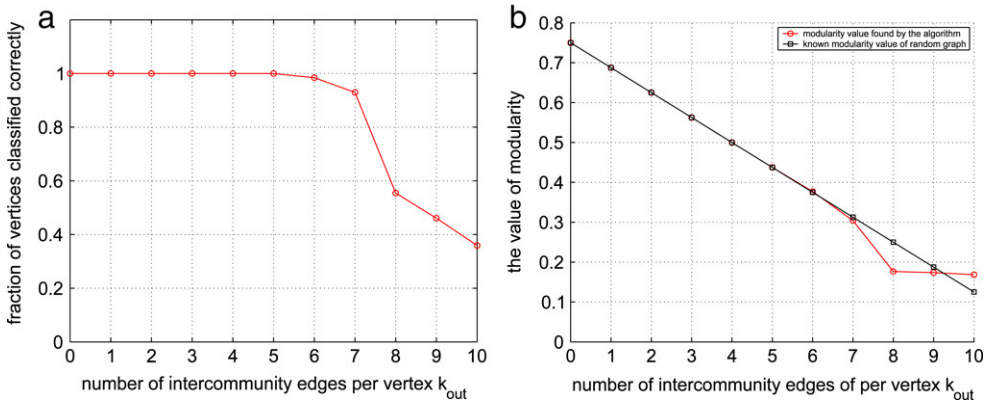

Fig. 2. (a) The fraction of vertices correctly classified in computer-generated graphs of the type described in the text, as the average number of intercommunity edges per vertex is varied. (b) Maximum modularity found by the algorithm (circle), compared to the real modularity (square).

5. Tests of the algorithm

One way that has been employed to test sensitivity in many cases is to see how well a particular method performs when applied to computer-generated random graphs with a well-known predetermined community structure [22]. We adopt this test method, using networks of 128 nodes which are grouped into four equal sized communities of size 32. Each node has an average degree $\langle k \rangle = 16$. Edges between two nodes are introduced with different probabilities depending on whether the two nodes belong to the same group or not: every node has k_{in} links on average to its fellows in the same community, and k_{out} links to the outer world, keeping $k_{\text{in}} + k_{\text{out}} = 16$.

This produces graphs that have known community structure, but which are essentially random in other respects. Feeding these graphs into our algorithm, we measured the fraction of the average number of intercommunity edges per vertex. The results are shown in Fig. 2(a). As the figure shows, the algorithm performs nearly perfectly when $k_{\text{out}} < 7$, classifying 95% or more of the vertices correctly. Only for $k_{\text{out}} \geqslant 7$ does the fraction correctly classified start to fall off substantially. In other words, the algorithm performs very well almost to the point at which each vertex has as much intercommunity as intracommunity connection.

For comparison, we also show in Fig. 2(b) the maximum modularity found by our algorithm (circle) and the real modularity of the known splitting of the computer-generated networks (square). We can see, for these computer-generated networks, that our algorithm generates excellent results. We notice that when $k_{\text{out}} > 9$, the modularity found by our algorithm is larger than the real modularity. This may be attributed to the fact that the community structure is blurred now, and so we can get communities through our algorithm better than those generated by computer.

6. Conclusions

In this paper we have introduced a new algorithm aimed at detecting community structure in complex networks. The method combines spectral techniques, a vector partition problem, and the concept of modularity and is a natural extension of bi-partitioning to multiple eigenvectors.

In practice, as Section 5 shows, the algorithm works very well, finding the known split of the network into four groups almost perfectly.

The weakest part of the method is that when K is get larger, the number of columns of the coarse assignment matrix X^* gets larger exponentially. But in practice, in all the cases studied, the best splitting is found with a relatively small number of eigenvectors. So we think that our method is a reliable and very efficient one.

References

[1] Steven H. Strogatz, Exploring complex networks, Nature 410 (2001) 268–276.
[2] Réka Albert, Albert-László Barabási, Statistical mechanics of complex networks, Rev. Modern Phys. 74 (2002) 47–91.
[3] M.E.J. Newman, The structure and function of complex networks, SIAM Rev. 45 (2003) 167.
[4] Aaron Clause, M.E.J. Newman, Cristopher Moore, Finding community structure in very large networks, Phys. Rev. E 70 (066111) (2004) 6.

[5] Santo Fortunato, Vito Latora, Massimo Marchiori, Method to find community structure based on information centrality, Phys. Rev. E 70 (056105) (2004) 13.
[6] C. Castellano, F. Cecconi, V. Loreto, D. Parisi, F. Radicchi, Self-contained algorithms to detect communities in networks, Eur. Phys. J. B 38 (2004) 311–319.
[7] Jordi Duch, Alex Arenas, Community detection in complex networks using extremal optimization, Phys. Rev. E 72 (027104) (2005) 4.
[8] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA 99 (12) (2002) 7821–7826.
[9] M.E.J. Newman, Fast algorithm for detecting community structure in networks, Phys. Rev. E 69 (066133) (2004) 5.
[10] A. Capocci, V.D.P. Servedio, G. Caldarelli, F. Colaiori, Detecting communities in large networks, Physica A 352 (2–4) (2005) 669–676.
[11] Filippo Radicchi, Claudio Castellano, Federico Cecconi, Vitorio Loreto, Domenico Parisi, Defining and identifying communities in networks, Proc. Natl. Acad. Sci. USA 101 (9) (2004) 2658–2663.
[12] Leon Danon, Albert Daz-Guilera, Jordi Duch, Alex Arenas, Comparing community structure identification, J. Stat. Mech. (2005) P09008.
[13] M.E.J. Newman, Detecting community structure in networks, Eur. Phys. J. B 38 (2) (2004) 321–330.
[14] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D.U. Hwang, Complex networks: Structure and dynamics, Phys. Rep. 424 (2006) 175–308.
[15] M.E.J. Newman, Finding community structure in networks using the eigenvectors of matrices, Phys. Rev. E 74 (036104) (2006) 19.
[16] M.E.J. Newman, From the cover: Modularity and community structure in networks, Proc. Natl. Acad. Sci. USA 103 (2006) 8577–8582.
[17] Scott White, Padhraic Smyth, A spectral clustering approach to finding communities in graphs, in: H. Kargupta, J. Srivastava, C. Kamath, A. Goodman (Eds.), 5th SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, Philadelphia, 2005, pp. 274–285.
[18] Luca Donetti, Miguel A. Munoz, Detecting network communities: A new systematic and efficient algorithm, J. Stat. Mech. (2004) P10012.
[19] Luca Donetti, Miguel A. Munoz, Improved spectral algorithm for the detection of network communities, 1 (0504059) arXiv:physics, 2005, p. 4.
[20] Andrew.J. Seary, William.D. Richards, Partitioning networks by eigenvectors, in: M.G. Everett, K. Rennolls (Eds.), Proceedings of the International Conference on Social Networks, in: Methodology, vol. 1, 1996, pp. 47–58.
[21] Charles J. Alpert, Andrew B. Kahng, So-Zen Yao, Spectral partitioning with multiple eigenvectors, Discrete Appl. Math. 90 (1–3) (1999) 3–26.
[22] M.E.J. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (026113) (2004) 15.